```
UUU         UUU  EEEEEEEEEEEEEEE  TTTTTTTTTTTTTTT  PPPPPPPPPPPP       SSSSSSSSSSS  YYY         YYY
UUU         UUU  EEEEEEEEEEEEEEE  TTTTTTTTTTTTTTT  PPPPPPPPPPPP       SSSSSSSSSSS  YYY         YYY
UUU         UUU  EEEEEEEEEEEEEEE  TTTTTTTTTTTTTTT  PPPPPPPPPPPP       SSSSSSSSSSS  YYY         YYY
UUU         UUU  EEE                    TTT        PPP         PPP  SSS              YYY         YYY
UUU         UUU  EEE                    TTT        PPP         PPP  SSS              YYY         YYY
UUU         UUU  EEE                    TTT        PPP         PPP  SSS              YYY         YYY
UUU         UUU  EEE                    TTT        PPP         PPP  SSS                 YYY   YYY
UUU         UUU  EEE                    TTT        PPP         PPP  SSS                 YYY   YYY
UUU         UUU  EEE                    TTT        PPP         PPP  SSS                 YYY   YYY
UUU         UUU  EEEEEEEEEEEE           TTT        PPPPPPPPPPPP       SSSSSSSSS           YYY
UUU         UUU  EEEEEEEEEEEE           TTT        PPPPPPPPPPPP       SSSSSSSSS           YYY
UUU         UUU  EEEEEEEEEEEE           TTT        PPPPPPPPPPPP       SSSSSSSSS           YYY
UUU         UUU  EEE                    TTT        PPP                       SSS           YYY
UUU         UUU  EEE                    TTT        PPP                       SSS           YYY
UUU         UUU  EEE                    TTT        PPP                       SSS           YYY
UUU         UUU  EEE                    TTT        PPP                       SSS           YYY
UUU         UUU  EEE                    TTT        PPP                       SSS           YYY
UUUUUUUUUUUUUUU  EEEEEEEEEEEEEEE        TTT        PPP              SSSSSSSSSSS           YYY
UUUUUUUUUUUUUUU  EEEEEEEEEEEEEEE        TTT        PPP              SSSSSSSSSSS           YYY
UUUUUUUUUUUUUUU  EEEEEEEEEEEEEEE        TTT        PPP              SSSSSSSSSSS           YYY
```

```
  SSSSSSSS    AAAAAA    TTTTTTTTTT   SSSSSSSS   SSSSSSSS   SSSSSSSS  44    44       11
  SSSSSSSS    AAAAAA    TTTTTTTTTT   SSSSSSSS   SSSSSSSS   SSSSSSSS  44    44       11
SS           AA    AA       TT     SS         SS         SS           44    44     1111
SS           AA    AA       TT     SS         SS         SS           44    44     1111
SS           AA    AA       TT     SS         SS         SS           44    44       11
SS           AA    AA       TT     SS         SS         SS           44    44       11
  SSSSSS     AA    AA       TT       SSSSSS     SSSSSS     SSSSSS   4444444444     11
  SSSSSS     AA    AA       TT       SSSSSS     SSSSSS     SSSSSS   4444444444     11
        SS   AAAAAAAAAA       TT             SS         SS         SS       44     11
        SS   AAAAAAAAAA       TT             SS         SS         SS       44     11
        SS   AA    AA         TT             SS         SS         SS       44     11          ....
        SS   AA    AA         TT             SS         SS         SS       44     11          ....
SSSSSSSS     AA    AA         TT     SSSSSSSS   SSSSSSSS   SSSSSSSS       44   111111        ....
SSSSSSSS     AA    AA         TT     SSSSSSSS   SSSSSSSS   SSSSSSSS       44   111111        ....


LL             IIIIII      SSSSSSSS
LL             IIIIII      SSSSSSSS
LL               II      SS
LL               II      SS
LL               II      SS
LL               II      SS
LL               II        SSSSSS
LL               II        SSSSSS
LL               II              SS
LL               II              SS
LL               II              SS
LL               II              SS
LLLLLLLLLL     IIIIII    SSSSSSSS
LLLLLLLLLL     IIIIII    SSSSSSSS
```

```
0000     1          .TITLE  SATSSS41 SATS SYSTEM SERVICE TESTS  $EXIT (SUCC S.C.)
0000     2          .IDENT  'V04-000'
0000     3
0000     4 ;
0000     5 ;*****************************************************************************
0000     6 ;*                                                                           *
0000     7 ;*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                  *
0000     8 ;*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                   *
0000     9 ;*  ALL RIGHTS RESERVED.                                                     *
0000    10 ;*                                                                           *
0000    11 ;*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED    *
0000    12 ;*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
0000    13 ;*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER    *
0000    14 ;*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY    *
0000    15 ;*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY    *
0000    16 ;*  TRANSFERRED.                                                             *
0000    17 ;*                                                                           *
0000    18 ;*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE    *
0000    19 ;*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
0000    20 ;*  CORPORATION.                                                             *
0000    21 ;*                                                                           *
0000    22 ;*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS    *
0000    23 ;*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                  *
0000    24 ;*                                                                           *
0000    25 ;*                                                                           *
0000    26 ;*****************************************************************************
0000    27 ;
0000    28
0000    29 ;++
0000    30 ; FACILITY:     SYSTST (SATS SYSTEM SERVICE TESTS)
0000    31 ;
0000    32 ; ABSTRACT:
0000    33 ;
0000    34 ;               THIS MODULE CONTAINS SUBROUTINES WHICH, WHEN LINKED
0000    35 ; WITH SUCCOMMON.OBJ, FORM TEST MODULE SATSSS41 TO TEST SUCCESSFUL
0000    36 ; OPERATION OF THE $EXIT SYSTEM SERVICE. THE SERVICE IS INVOKED
0000    37 ; UNDER VARIOUS INPUT CONDITIONS WITH VARYING INPUT PARAMETERS. ONLY
0000    38 ; SUCCESSFUL STATUS CODES ARE EXPECTED IN THIS TEST MODULE.  CORRECT
0000    39 ; OPERATION OF THE SERVICE FOR EACH OF ITS ISSUANCES IS VERIFIED BY
0000    40 ; CHECKING FOR AN SS$_NORMAL STATUS CODE, EXPECTED RETURN ARGUMENTS
0000    41 ; AND EXPECTED FUNCTIONALITY PERFORMED.
0000    42 ;
0000    43 ; ENVIRONMENT:  USER MODE IMAGE; NEEDS CMKRNL PRIVILEGE,
0000    44 ;               DYNAMICALLY ACQUIRES OTHER PRIVILEGES, AS NEEDED.
0000    45 ;
0000    46 ; AUTHOR: THOMAS L. CAFARELLA,          CREATION DATE: OCT, 1977
0000    47 ;
0000    48 ; MODIFIED BY:
0000    49 ;
0000    50 ;       V03-001 LDJ0001         Larry D. Jones,         23-Jun-1983
0000    51 ;               Removed quota list to use default sysboot quota values.
0000    52 ;--
```

```
                                N 2
0000     54          .SBTTL  DECLARATIONS
0000     55 ;
0000     56 ; INCLUDE FILES:
0000     57 ;
0000     58          $PRVDEF                         ; PRIVILEGE BIT DEFINITIONS
0000     59          $PHDDEF                         ; PROCESS HEADER OFFSETS
0000     60          $PQLDEF                         ; PROCESS QUOTA CODES
0000     61          $PCBDEF                         ; PCB LABELS
0000     62          $DIBDEF                         ; DEVICE INFO BLOCK OFFSETS
0000     63 ;
0000     64 ; MACROS:
0000     65 ;
0000     66 ;
0000     67 ; EQUATED SYMBOLS:
0000     68 ;
0000     69 ;
0000     70 ; OWN STORAGE:
0000     71 ;
```

```
00000000     73          .PSECT  RODATA,RD,NOWRT,NOEXE,LONG
     0000    74 TEST_MOD_NAME:: STRING  C,<SATSSS41>     ; TEST MODULE NAME
     0009    75 TEST_MOD_NAME_D: STRING I,<SATSSS41>     ; TEST MODULE NAME DESCRIPTOR
     0019    76 MSG1_INP_CTL:   STRING  I,<  SSEXI!4ZW: CONDITIONS:>
     0039    77                                          ; FAO CTL STRING FOR MSG1 IN SUCCOMMON.MAR
     0039    78 MSG3_ERR_CTL::  STRING  I,< *SSEXI!4ZW: !AS>
     0051    79                                          ; FAO CTL STRING FOR MSG3 IN SUCCOMMON.MAR
     0051    80 CRENAME:        STRING  I,<SATSSS41_CRE> ; PROCESS & MBX NAME FOR CREATED PROCESS
     0065    81 IMAGNAM:        STRING  I,<SYSTST$RES:SATSUT09.EXE>
     0084    82                                          ; IMAGE NAME FOR CREATED PROCESS
     0084    83 ;QUOTALIST:     $QUOTA  CPULM,0          ; INFINITE CPU
     0084    84 ;               $QUOTA  BYTLM,512        ; BYTE LIMIT FOR BUFFERED I/O
     0084    85 ;               $QUOTA  FILLM,2          ; OPEN FILE COUNT LIMIT
     0084    86 ;               $QUOTA  PGFLQUOTA,10     ; PAGING FILE QUOTA
     0084    87 ;               $QUOTA  PRCLM,2          ; SUBPROCESS QUOTA
     0084    88 ;               $QUOTA  TQELM,3          ; TIMER QUEUE ENTRY QUOTA
     0084    89 ;               $QUOTA  LISTEND          ; DEFINES END OF LIST
```

```
                00000000     91          .PSECT  RWDATA,RD,WRT,NOEXE,LONG
00000008  0000  00000008     92 PRIVMASK:        .BLKQ   1               ; ADDR OF PRIVILEGE MASK (IN PHD)
0000000C  0008  0000000C     93 MBXCHAN:         .BLKL   1               ; CHAN NO. FOR MAILBOX FOR CREATED PROCESS
          000C  00000010     94 MBXCHANINFO:                             ; CHANNEL INFO RETURNED BY GETCHN
00000074  000C  00000074     95                  .LONG   DIB$K_LENGTH
00000014' 0010  00000078     96                  .ADDRESS .+4
00000088  0014  0000008C     97                  .BLKB   DIB$K_LENGTH
0000008C  0088  00000110     98 MBXUNIT:         .BLKL   1               ; SAVE AREA FOR MAILBOX UNIT NUMBER
          008C  00000114     99 MBXBUFF:         STRING  0,120           ; MAILBOX BUFFER FOR CREATED PROCESS
00000110  010C  00000190    100 CREPID:          .BLKL   1               ; PID OF CREATED PROCESS
```

```
                        0110   102          .SBTTL  CONDITION TABLES
                        0110   103 ;
                        0110   104 ;        ***** CONDITION TABLES FOR EXIT SYSTEM SERVICE *****
                        0110   105 ;
                        0110   106          COND    1,NOTARG,<PROCESS TYPE>,-
                        0110   107                    <SUBPROCESS>,-
                        0110   108                    <DETACHED, DIFFERENT GROUP>,-
                        0110   109                    <DETACHED, SAME GROUP, SAME MEMBER>,-
                        0110   110                    <DETACHED, SAME GROUP, DIFFERENT MEMBER>,-
                        0110   111
00000000                019C   112                      .LONG       0          ; PSEUDO-UIC
000001A4                01A0   113                      .BLKL       1          ; UIC
000001AB                01A4   114                      .BLKL       1          ; UIC
000001AC                01A8   115                      .BLKL       1          ; UIC
                        01AC   116 ;
                        01AC   117          COND    2,NULL
                        01AD   118
                        01AD   119          COND    3,NULL
                        01AE   120
                        01AE   121          COND    4,NULL
                        01AF   122
                        01AF   123          COND    5,NULL
                        01B0   124
                    00000000   125          .PSECT  SATSSS41,RD,WRT,EXE
```

```
                                        .SBTTL  TM_SETUP, TM_CLEANUP
                                ;++
                                ; FUNCTIONAL DESCRIPTION:
                                ;
                                ;               TM_SETUP AND TM_CLEANUP ARE CALLED TO PERFORM
                                ; REQUIRED HOUSEKEEPING AT THE BEGINNING AND END, RESPECTIVELY, OF
                                ; TEST MODULE EXECUTION.
                                ;
                                ; CALLING SEQUENCE:
                                ;
                                ;       BSBW TM_SETUP   BSBW TM_CLEANUP
                                ;
                                ; INPUT PARAMETERS:
                                ;
                                ;       NONE
                                ;
                                ; IMPLICIT INPUTS:
                                ;
                                ;       NONE
                                ;
                                ; OUTPUT PARAMETERS:
                                ;
                                ;       NONE
                                ;
                                ; IMPLICIT OUTPUTS:
                                ;
                                ;       TM_SETUP:  COND TABLE INDEX REGISTERS (R2,3,4,5,6) CLEARED;
                                ;                  ALL PRIVILEGES ACQUIRED.
                                ;
                                ; COMPLETION CODES:
                                ;
                                ;       EFLAG SET TO NON-ZERO IF ERROR ENCOUNTERED.
                                ;
                                ; SIDE EFFECTS:
                                ;
                                ;       SS_CHECK AND ERR_EXIT MACROS CAUSE PREMATURE EXIT
                                ;       (VIA RSB) IF ERROR ENCOUNTERED.
                                ;
                                ;--
                                
                                TM_SETUP::
                        52  D4    CLRL    R2                      ; INITIALIZE
                        53  D4    CLRL    R3                      ; .. CONDITION
                        54  D4    CLRL    R4                      ; .... TABLE
                        55  D4    CLRL    R5                      ; ...... INDEX
                        56  D4    CLRL    R6                      ; ........ REGISTERS
                    FFF3'   30    BSBW    MOD_MSG_PRINT           ; PRINT TEST MODULE BEGIN MSG
00000000'EF  00000000'EF   DE    MOVAL   TEST_MOD_SUCC,TMD_ADDR  ; ASSUME END MSG WILL SHOW SUCCESS
   03  00  00000000'8F   F0    INSV    #SUCCESS,#0,#3,MOD_MSG_CODE ; ADJUST STATUS CODE FOR SUCCESS
           00000000'EF
                                MODE    TO,S$,KRNL              ; KERNEL MODE TO ACCESS PHD
       59  00000000'9F   D0    MOVL    @#CTL$GL_PHD,R9         ; GET PROCESS HEADER ADDRESS
       00000000'EF  69   DE    MOVAL   PHD$Q_PRIVMSK(R9),PRIVMASK ; GET PRIV MASK ADDRESS
                                MODE    FROM,S$ ; BACK TO USER MODE
                                PRIV    ADD,ALL                 ; GET ALL PRIVILEGES
```

```
                        0077   183          $SETPRN_S TEST_MOD_NAME_D       ; SET PROCESS NAME
                        0084   184          SS_CHECK  NORMAL                ; CHECK STATUS CODE RETURNED FROM SETPRN
                        00B2   185 ;
                        00B2   186 ; THE FOLLOWING CODE ESTABLISHES UIC'S IN THE CONDITION 1 TABLE
                        00B2   187 ;
                        00B2   188          MODE    TO,20$,KRNL             ; KERNEL MODE TO ACCESS PCB
      59  00000000'9F   D0  00D5   189          MOVL    @#SCH$GL_CURPCB,R9      ; GET CURRENT PCB ADDRESS
          59  00BC C9   D0  00DC   190          MOVL    PCB$L_UIC(R9),R9        ; PICK UP UIC FROM PCB
                        00E1   191          MODE    FROM,20$                ; ... AND GET BACK TO USER MODE
                        00E2   192 ;
                        00E2   193 ; R9 NOW CONTAINS "MY" UIC
                        00E2   194 ;
            5A    01    9A  00E2   195          MOVZBL  #1,R10                  ; GET COND1 TABLE INDEX NUMBER INTO A REG
      59  00010000 8F    C1  00E5   196          ADDL3   #^X10000,R9,COND1_E[R10] ; PUT DIFF GROUP UIC INTO 2ND TABLE ELT
        0000019C'EF4A
                        00EC
            5A    D6  00F2   197          INCL    R10                     ; POINT TO 3RD COND1 TABLE ELEMENT
  0000019C'EF4A   59    D0  00F4   198          MOVL    R9,COND1_E[R10]         ; PUT MY UIC INTO TABLE
            5A    D6  00FC   199          INCL    R10                     ; POINT TO 4TH COND1 TABLE ELEMENT
0000019C'EF4A   59    01    C1  00FE   200          ADDL3   #1,R9,COND1_E[R10]      ; PUT DIFF MEMBER UIC INTO THE TABLE
                        0107   201          $CREMBX_S CHAN=MBXCHAN, LOGNAM=CRENAME, - ; GET MAILBOX FOR PROCESS
                        0107   202                  MAXMSG=#120, PROMSK=#0, BUFQUO=#240
                        012C   203          SS_CHECK NORMAL                 ; CHECK NORMAL COMPLETION
                        015A   204          $GETCHN_S CHAN=MBXCHAN, -       ; GET CHAN INFO (UNIT NUMBER)
                        015A   205                  PRIBUF=MBXCHANINFO
                        0174   206          SS_CHECK NORMAL                 ; CHECK NORMAL COMPLETION
00000088'EF   00000020'EF   3C  01A2   207          MOVZWL  MBXCHANINFO+8+DIB$W_UNIT,MBXUNIT ; SAVE MAILBOX UNIT NUMBER
                        05  01AD   208          RSB                             ; RETURN TO MAIN ROUTINE
                        01AE   209 TM_CLEANUP::
                        01AE   210          $DELMBX_S  MBXCHAN              ; DELETE TERMINATION MAILBOX
                FE41'   30  01BC   211          BSBW    MOD_MSG_PRINT           ; PRINT TEST MODULE END MSG
                        05  01BF   212          RSB                             ; RETURN TO MAIN ROUTINE
```

```
                        .SBTTL  CONDITION SUBROUTINES - SETUP AND CLEANUP
                ;++
                ; FUNCTIONAL DESCRIPTION:
                ;
                ;               CONDX AND CONDX CLEANUP ARE SUBROUTINES WHICH ARE EXECUTED
                ; BEFORE AND AFTER THE VERIFY SUBROUTINE, RESPECTIVELY, WHENEVER A NEW
                ; CONDITION X VALUE IS SELECTED (SEE FUNCTIONAL DESCRIPTION OF SUCCOMMON
                ; ROUTINE IN SUCCOMMON.MAR). ANY SETUP FUNCTION PARTICULAR TO THE
                ; CONDITION X TABLE IS INCLUDED IN THE CONDX SUBROUTINE AND CLEANED
                ; UP, IF NECESSARY, IN THE CONDX_CLEANUP SUBROUTINE. THIS INCLUDES,
                ; ESPECIALLY, CODE TO DETECT CONFLICTS AMONG CURRENT ENTRIES IN TWO
                ; OR MORE CONDITION TABLES. IF A CONFLICT IS DETECTED, A NON-ZERO
                ; VALUE IS STORED INTO CONFLICT, WHICH CAUSES THE CALLING ROUTINE
                ; (SUCCOMMON) TO SKIP THE CURRENT ENTRY IN THE CONDITION X TABLE.
                ;
                ; CALLING SEQUENCE:
                ;
                ;       BSBW CONDX   BSBW CONDX_CLEANUP
                ;         WHERE X = 1,2,3,4,5
                ;
                ; INPUT PARAMETERS:
                ;
                ;       CONFLICT = 0
                ;
                ; IMPLICIT INPUTS:
                ;
                ;       R2,3,4,5,6 CONTAIN CURRENT CONDITION TABLE INDEX VALUES
                ;         FOR COND TABLES 1,2,3,4,5, RESPECTIVELY.
                ;
                ; OUTPUT PARAMETERS:
                ;
                ;       CONFLICT SET TO NON-ZERO IF COND TABLE CONFLICT DETECTED.
                ;
                ; IMPLICIT OUTPUTS:
                ;
                ;       R2,3,4,5,6 PRESERVED
                ;
                ; COMPLETION CODES:
                ;
                ;       NONE
                ;
                ; SIDE EFFECTS:
                ;
                ;       NONE
                ;
                ;--


                COND1::
05              RSB                             ; RETURN TO MAIN ROUTINE
                COND1_CLEANUP::
05              RSB                             ; RETURN TO MAIN ROUTINE
                COND2::
05              RSB                             ; RETURN TO MAIN ROUTINE
                COND2_CLEANUP::
05              RSB                             ; RETURN TO MAIN ROUTINE
```

```
                     01C4    271 COND3::
 05                  01C4    272         RSB                             ; RETURN TO MAIN ROUTINE
                     01C5    273 COND3_CLEANUP::
 05                  01C5    274         RSB                             ; RETURN TO MAIN ROUTINE
                     01C6    275 COND4::
 05                  01C6    276         RSB                             ; RETURN TO MAIN ROUTINE
                     01C7    277 COND4_CLEANUP::
 05                  01C7    278         RSB                             ; RETURN TO MAIN ROUTINE
                     01C8    279 COND5::
 05                  01C8    280         RSB                             ; RETURN TO MAIN ROUTINE
                     01C9    281 COND5_CLEANUP::
 05                  01C9    282         RSB                             ; RETURN TO MAIN ROUTINE
```

```
                                                     .SBTTL  FORM_CONDS
                                                ;++
                                                ; FUNCTIONAL DESCRIPTION:
                                                ;
                                                ;               FORM_CONDS FORMATS AND PRINTS INFORMATION ABOUT
                                                ;  THE CURRENT ELEMENT IN EACH OF THE CONDITION TABLES.
                                                ;
                                                ; CALLING SEQUENCE:
                                                ;
                                                ;       BSBW FORM_CONDS
                                                ;
                                                ; INPUT PARAMETERS:
                                                ;
                                                ;       NONE
                                                ;
                                                ; IMPLICIT INPUTS:
                                                ;
                                                ;       R2,3,4,5,6 CONTAIN CURRENT CONDITION TABLE INDEX VALUES
                                                ;         FOR COND TABLES 1,2,3,4,5, RESPECTIVELY.
                                                ;       FOR X = 1,2,3,4,5 :
                                                ;               CONDX_T - TITLE TEXT FOR CONDX TABLE
                                                ;               CONDX_TAB - ELEMENT TEXT FOR CONDX TABLE
                                                ;               CONDX_C - CONTEXT OF THE CONDX TABLE
                                                ;               CONDX_E - DATA ELEMENTS OF THE CONDX TABLE
                                                ;
                                                ; OUTPUT PARAMETERS:
                                                ;
                                                ;       NONE
                                                ;
                                                ; IMPLICIT OUTPUTS:
                                                ;
                                                ;       NONE
                                                ;
                                                ; COMPLETION CODES:
                                                ;
                                                ;       NONE
                                                ;
                                                ; SIDE EFFECTS:
                                                ;
                                                ;       NONE
                                                ;
                                                ;--
                                                FORM_CONDS::
                                                        $FAO_S  MSG1_INP_CTL,FAO_LEN,FAO_DESC,TESTNUM
                                                                                        ; FORMAT CONDITIONS HEADER MSG
                                                        BSBW    OUTPUT_MSG              ; ... AND PRINT IT
                                                        CMPB    #COND1_C,#NULL          ; IS CONDITION 1 NULL ?
                                                        BNEQU   10$                     ; NO -- CONTINUE
                                                        BRW     FORM_CONDSX             ; YES -- SUBROUTINE IS FINISHED
                                                10$:
                                                        MOVAL   COND1_T,MSG_A           ; SAVE ADDRESS OF CONDITION 1 TITLE FOR FAO
                                                        MOVL    COND1_TAB[R2],MSG_B     ; SAVE ADDR OF COND 1 CURR TEXT ELT FOR FAO
                                                        MOVB    #COND1_C,MSG_CTXT       ; SAVE CONDITION 1 CONTEXT FOR FAO
                                                        MOV_VAL COND1_C,COND1_E[R2],MSG_DATA1 ; GIVE COND 1 DATA VALUE TO FAO
```

```
                                                     J 3
                     FDEB'   30  0212   341          BSBW    WRITE_MSG2                      ; FORMAT AND WRITE CONDITION 1 MSG
                  14   14   91  0215   342          CMPB    #COND2_C,#NULL                  ; IS CONDITION 2 NULL ?
                       03   12  0218   343          BNEQU   20$                             ; NO -- CONTINUE
                     0096   31  021A   344          BRW     FORM_CONDSX                     ; YES -- SUBROUTINE IS FINISHED
                            021D   345 20$:
  00000000'EF    000001AC'EF   DE  021D   346          MOVAL   COND2_T,MSG_A                   ; SAVE ADDRESS OF CONDITION 2 TITLE FOR FAO
00000000'EF    000001AC'EF43   D0  0228   347          MOVL    COND2_TAB[R3],MSG_B             ; SAVE ADDR OF COND 2 CURR TEXT ELT FOR FAO
          00000000'EF    14   90  0234   348          MOVB    #COND2_C,MSG_CTXT               ; SAVE CONDITION 2 CONTEXT FOR FAO
                            023B   349          MOV_VAL COND2_C,COND2_E[R3],MSG_DATA1 ; GIVE COND 2 DATA VALUE TO FAO
                     FDC2'   30  023B   350          BSBW    WRITE_MSG2                      ; FORMAT AND WRITE CONDITION 2 MSG
                  14   14   91  023E   351          CMPB    #COND3_C,#NULL                  ; IS CONDITION 3 NULL ?
                       03   12  0241   352          BNEQU   30$                             ; NO -- CONTINUE
                     006D   31  0243   353          BRW     FORM_CONDSX                     ; YES -- SUBROUTINE JS FINISHED
                            0246   354 30$:
  00000000'EF    000001AD'EF   DE  0246   355          MOVAL   COND3_T,MSG_A                   ; SAVE ADDRESS OF CONDITION 3 TITLE FOR FAO
00000000'EF    000001AD'EF44   D0  0251   356          MOVL    COND3_TAB[R4],MSG_B             ; SAVE ADDR OF COND 3 CURR TEXT ELT FOR FAO
          00000000'EF    14   90  025D   357          MOVB    #COND3_C,MSG_CTXT               ; SAVE CONDITION 3 CONTEXT FOR FAO
                            0264   358          MOV_VAL COND3_C,COND3_E[R4],MSG_DATA1 ; GIVE COND 3 DATA VALUE TO FAO
                     FD99'   30  0264   359          BSBW    WRITE_MSG2                      ; FORMAT AND WRITE CONDITION 3 MSG
                  14   14   91  0267   360          CMPB    #COND4_C,#NULL                  ; IS CONDITION 4 NULL ?
                       47   13  026A   361          BEQLU   FORM_CONDSX                     ; YES -- SUBROUTINE IS FINISHED
  00000000'EF    000001AE'EF   DE  026C   362          MOVAL   COND4_T,MSG_A                   ; SAVE ADDRESS OF CONDITION 4 TITLE FOR FAO
00000000'EF    000001AE'EF45   D0  0277   363          MOVL    COND4_TAB[R5],MSG_B             ; SAVE ADDR OF COND 4 CURR TEXT ELT FOR FAO
          00000000'EF    14   90  0283   364          MOVB    #COND4_C,MSG_CTXT               ; SAVE CONDITION 4 CONTEXT FOR FAO
                            028A   365          MOV_VAL COND4_C,COND4_E[R5],MSG_DATA1 ; GIVE COND 4 DATA VALUE TO FAO
                     FD73'   30  028A   366          BSBW    WRITE_MSG2                      ; FORMAT AND WRITE CONDITION 4 MSG
                  14   14   91  028D   367          CMPB    #COND5_C,#NULL                  ; IS CONDITION 5 NULL ?
                       21   13  0290   368          BEQLU   FORM_CONDSX                     ; YES -- SUBROUTINE IS FINISHED
  00000000'EF    000001AF'EF   DE  0292   369          MOVAL   COND5_T,MSG_A                   ; SAVE ADDRESS OF CONDITION 5 TITLE FOR FAO
00000000'EF    000001AF'EF46   D0  029D   370          MOVL    COND5_TAB[R6],MSG_B             ; SAVE ADDR OF COND 5 CURR TEXT ELT FOR FAO
          00000000'EF    14   90  02A9   371          MOVB    #COND5_C,MSG_CTXT               ; SAVE CONDITION 5 CONTEXT FOR FAO
                            02B0   372          MOV_VAL COND5_C,COND5_E[R6],MSG_DATA1 ; GIVE COND 5 DATA VALUE TO FAO
                     FD4D'   30  02B0   373          BSBW    WRITE_MSG2                      ; FORMAT AND WRITE CONDITION 5 MSG
                            02B3   374 FORM_CONDSX:
                       05   C2B3   375          RSB                                     ; RETURN TO CALLER
```

```
02B4    377          .SBTTL  VERIFY
02B4    378 ;++
02B4    379 ; FUNCTIONAL DESCRIPTION:
02B4    380 ;
02B4    381 ;               VERIFY IS CALLED ONCE FOR EACH COMBINATION OF CONDITION
02B4    382 ; TABLE VALUES (AS DETERMINED BY THE INDEX REGISTERS R2,3,4,5,6 FOR
02B4    383 ; COND TABLES 1,2,3,4,5, RESPECTIVELY). VERIFY ESTABLISHES THE CONDITIONS
02B4    384 ; SPECIFIED BY THE COND TABLES AND ISSUES THE SUBJECT SYSTEM SERVICE
02B4    385 ; ($EXIT). THEN, THE SUCCESSFUL OPERATION OF THE SERVICE IS VERIFIED
02B4    386 ; BY EXAMINING THE STATUS CODE RETURNED, THE VALUES FOR RETURN ARGUMENTS
02B4    387 ; AND THE FUNCTIONALITY PERFORMED. THE EXAMINATIONS TAKE THE FORM OF
02B4    388 ; COMPARISONS AGAINST EXPECTED VALUES. ANY FAILING COMPARISON CAUSES AN
02B4    389 ; ERR_EXIT MACRO TO BE EXECUTED (EITHER DIRECTLY, OR INDIRECTLY,
02B4    390 ; THROUGH THE SS_CHECK MACRO); ERR_EXIT SETS EFLAG TO NON-ZERO,
02B4    391 ; PRINTS ERROR MESSAGES AND CAUSES AN IMMEDIATE RSB TO CALLER.
02B4    392 ; WHEN ERR_EXIT IS EXECUTED, FURTHER CALLS TO VERIFY ARE SUPPRESSED,
02B4    393 ; AND, AFTER EXECUTING CLEANUP SUBROUTINES, THE IMAGE EXITS.
02B4    394 ;
02B4    395 ; CALLING SEQUENCE:
02B4    396 ;
02B4    397 ;       BSBW VERIFY
02B4    398 ;
02B4    399 ; INPUT PARAMETERS:
02B4    400 ;
02B4    401 ;       NONE
02B4    402 ;
02B4    403 ; IMPLICIT INPUTS:
02B4    404 ;
02B4    405 ;       R2,3,4,5,6 CONTAIN CURRENT CONDITION TABLE INDEX VALUES
02B4    406 ;         FOR COND TABLES 1,2,3,4,5, RESPECTIVELY.
02B4    407 ;       FOR X = 1,2,3,4,5 :
02B4    408 ;               CONDX_E - ADDRESS OF TABLE OF DATA VALUES FOR CONDX
02B4    409 ;                 TABLE. IF THE CONTEXT OF TABLE X IS A SYSTEM SERVICE
02B4    410 ;                 ARGUMENT, THE ARGUMENT NAME MAY BE USED AS A SYNONYM
02B4    411 ;                 FOR CONDX_E.
02B4    412 ;
02B4    413 ; OUTPUT PARAMETERS:
02B4    414 ;
02B4    415 ;       NONE
02B4    416 ;
02B4    417 ; IMPLICIT OUTPUTS:
02B4    418 ;
02B4    419 ;       VERIFY HAS NO OUTPUT. SINCE ITS PURPOSE IS TO TEST FOR ERRORS,
02B4    420 ;       IT MERELY RETURNS TO CALLER NORMALLY AFTER THE TESTS, PROVIDING
02B4    421 ;       ALL WERE SUCCESSFUL; IF AN ERROR IS DISCOVERED, RETURN IS VIA
02B4    422 ;       AN ERR_EXIT OR SS_CHECK MACRO, BOTH OF WHICH DOCUMENT DETECTED
02B4    423 ;       ERRORS.
02B4    424 ;
02B4    425 ; COMPLETION CODES:
02B4    426 ;
02B4    427 ;       EFLAG SET TO NON-ZERO IF ERROR ENCOUNTERED.
02B4    428 ;
02B4    429 ; SIDE EFFECTS:
02B4    430 ;
02B4    431 ;       SS_CHECK AND ERR_EXIT MACROS CAUSE PREMATURE EXIT
02B4    432 ;       (VIA RSB) IF ERROR ENCOUNTERED.
02B4    433 ;
```

```
                        02B4    434 ;--
                        02B4    435
                        02B4    436
                        02B4    437
                        02B4    438 VERIFY::
00000000'EF   95        02B4    439         TSTB    CFLAG                   ; SHOULD CONDITIONS BE PRINTED ?
          03  13        02BA    440         BEQL    5$                      ; NO -- CONTINUE
        FF0B  30        02BC    441         BSBW    FORM_CONDS              ; YES -- FMT & PRINT ALL CONDS FOR THIS T.C.
                        02BF    442 5$:
0000010C'EF   D4        02BF    443         CLRL    CREPID                  ; INDICATE CREATED PROCESS NOT YET CREATED
                        02C5    444         $CREPRC_S PIDADR=CREPID, PRCNAM=CRENAME, -
                        02C5    445                   UIC=COND1_E[R2], IMAGE=IMAGNAM, -
                        02C5    446                   MBXUNT=MBXUNIT,, QUOTA=QUOTALIST
                        02FC    447                                           ; CREATE THE SUBJECT PROCESS
                        02FC    448         SS_CHECK NORMAL                   ; ... AND MAKE SURE IT CREATED OK
                        032A    449         $QIOW_S CHAN=MBXCHAN, FUNC=#IO$_READVBLK, -
                        032A    450                 P1=MBXBUFF+8, P2=MBXBUFF
                        0353    451                                           ; WAIT FOR CREATED PROCESS TO SEND MAIL
                        0353    452         SS_CHECK NORMAL                   ; CHECK FOR NORMAL STATUS CODE
0000010C'EF 00000098'EF D1      0381    453         CMPL    MBXBUFF+12,CREPID ; DID CREATED PROC RETURN EXPECTED STATUS ?
          69  13        038C    454         BEQLU   VERIFYX                   ; YES -- ALL IS OK
00000000'EF 0000010C'EF D0      038E    455         MOVL    CREPID,EXPV       ; NO -- LOAD UP EXPECTED AND
00000000'EF 00000098'EF D0      0399    456         MOVL    MBXBUFF+12,RECV   ; ... RECEIVED VALUES, THEN EXIT
                        03A4    457         ERR_EXIT LONG,<INCORRECT EXIT STATUS CODE RETURNED IN MAILBOX>
                        03F7    458 VERIFYX:
                    05  03F7    459         RSB                               ; RETURN TO CALLER
```

```
                        .SBTTL  VFY_CLEANUP
                ;++
                ; FUNCTIONAL DESCRIPTION:
                ;
                ;               VFY_CLEANUP EXECUTES SYSTEM SERVICES TO UNDO THE
                ; EFFECT OF THOSE ISSUED IN THE VERIFY SUBROUTINE. VFY_CLEANUP MUST
                ; ASSUME THAT VERIFY MAY NOT HAVE EXECUTED IN ITS ENTIRETY (IF AN
                ; ERROR IS FOUND). ALSO, VFY_CLEANUP MAY ISSUE SS_CHECK OR ERR_EXIT
                ; ONLY AFTER PERFORMING ALL OF ITS CLEANUP OPERATIONS; THIS IS REQUIRED
                ; IN THE EVENT THAT VFY_CLEANUP IS CALLED DURING ERROR PROCESSING,
                ; WHEN PERFORMING THE REQUIRED CLEANUP IS MORE IMPORTANT THAN
                ; POSSIBLY DISCOVERING A SECOND ERROR.
                ;
                ; CALLING SEQUENCE:
                ;
                ;       BSBW VFY_CLEANUP
                ;
                ; INPUT PARAMETERS:
                ;
                ;       NONE
                ;
                ; IMPLICIT INPUTS:
                ;
                ;       R2,3,4,5,6 CONTAIN CURRENT CONDITION TABLE INDEX VALUES
                ;         FOR COND TABLES 1,2,3,4,5, RESPECTIVELY.
                ;       FOR X = 1,2,3,4,5 :
                ;               CONDX_E - ADDRESS OF TABLE OF DATA VALUES FOR CONDX
                ;                 TABLE. IF THE CONTEXT OF TABLE X IS A SYSTEM SERVICE
                ;                 ARGUMENT, THE ARGUMENT NAME MAY BE USED AS A SYNONYM
                ;                 FOR CONDX_E.
                ;
                ; OUTPUT PARAMETERS:
                ;
                ;       NONE
                ;
                ; IMPLICIT OUTPUTS:
                ;
                ;       NONE
                ;
                ; COMPLETION CODES:
                ;
                ;       EFLAG SET TO NON-ZERO IF ERROR ENCOUNTERED.
                ;
                ; SIDE EFFECTS:
                ;
                ;       SS_CHECK AND ERR_EXIT MACROS CAUSE PREMATURE EXIT
                ;       (VIA RSB) IF ERROR ENCOUNTERED.
                ;
                ;--


                VFY_CLEANUP::
05              RSB                             ; RETURN TO CALLER
                .END
```

```
$$$$                = 000003AE R     04
$$$CHARS            = 0000002E
$$$CHARS1           = 0000000A
$$$CHARS2           = 00000019
$$$CHARS3           = 00000021
$$$CHARS4           = 00000026
$$$CHARS5           = 00000000
$$$COND_A           = 00000003
$$$STRINGS          = 00000001
$$$STRINGS2         = 00000005
$$T1                = 00000001
$$T2                = 00000004
BYTE                = 00000001   G
CFLAG                ********    X   04
CHMRTN               ********    X   04
CHM_CONT             ********    X   04
COMP_SC              ********    X   04
CONDT                000001C0 RG     04
COND1_C             = 00000000
COND1_CLEANUP        000001C1 RG     04
COND1_E              0000019C R      03
COND1_H              0000011D RG     03
COND1_T              00000110 R      03
COND1_TAB            0000011E R      03
COND2                000001C2 RG     04
COND2_C             = 00000014
COND2_CLEANUP        000001C3 RG     04
COND2_H              000001AC RG     03
COND2_T              000001AC R      03
COND2_TAB            000001AC R      03
COND3                000001C4 RG     04
COND3_C             = 00000014
COND3_CLEANUP        000001C5 RG     04
COND3_H              000001AD RG     03
COND3_T              000001AD R      03
COND3_TAB            000001AD R      03
COND4                000001C6 RG     04
COND4_C             = 00000014
COND4_CLEANUP        000001C7 RG     04
COND4_H              000001AE RG     03
COND4_T              000001AE R      03
COND4_TAB            000001AE R      03
COND5                000001C8 RG     04
COND5_C             = 00000014
COND5_CLEANUP        000001C9 RG     04
COND5_H              000001AF RG     03
COND5_T              000001AF R      03
COND5_TAB            000001AF R      03
CRENAME              00000051 R      02
CREPID               0000010C R      03
CTL$GL_PHD           ********     X  04
DESC                = 00000010   G
DIB$K_LENGTH        = 00000074
DIB$W_UNIT          = 0000000C
EFLAG                ********    X   04
EXPV                 ********    X   04
FAO_DESC             ********    X   04
FAO_LEN              ********    X   04
FORM_CONDS           000001CA RG     04
FORM_CONDSX          000002B3 R      04
IMAGNAM              00000065 R      02
IO$_READVBLK         ********    X   04
LONG                = 00000004   G
MBXBUFF              0000008C R      03
MBXCHAN              00000008 R      03
MBXCHANINFO          0000000C R      03
MBXUNIT              00000088 R      03
MOD_MSG_CODE         ********    X   04
MOD_MSG_PRINT        ********    X   04
MSGT_INP_CTL         00000019 R      02
MSG3_ERR_CTL         00000039 RG     02
MSG_A                ********    X   04
MSG_B                ********    X   04
MSG_CTXT             ********    X   04
NOTARG              = 00000000   G
NULL                = 00000014   G
OUTPUT_MSG           ********    X   04
PCB$L_PIC           = 000000BC
PCV                  ********    X   04
PHD$Q_PRIVMSK       = 00000000
PRIVMASK             00000000 R      03
PRIV_ARGS           = 00000002
PROCESS_ERR          ********    X   04
QUAD                = 00000008   G
RECV                 ********    X   04
REST_REGS            ********    X   04
SAVE_REGS            ********    X   04
SCH$GL_CURPCB        ********    X   04
SS$_NORMAL           ********    X   04
SUCCESS              ********    X   04
SYS$CMKRNL           ********   GX   04
SYS$CREMBX           ********   GX   04
SYS$CREPRC           ********   GX   04
SYS$DELMBX           ********   GX   04
SYS$FAO              ********    X   04
SYS$GETCHN           ********   GX   04
SYS$QIOW             ********   GX   04
SYS$SETPRN           ********   GX   04
SYS$SETPRV           ********   GX   04
TESTNUM              ********    X   04
TEST_MOD_NAME        00000000 RG     02
TEST_MOD_NAME_D      00000009 R      02
TEST_MOD_SUCC        ********    X   04
TMD_ADDR             ********    X   04
TM_CLEANUP           000001AE RG     04
TM_SETUP             00000000 RG     04
VERIFY               000002B4 RG     04
VERIFYX              000003F7 R      04
VFY_CLEANUP          000003F8 RG     04
WORD                = 00000002   G
WRITE_MSG2           ********    X   04
```

Psect synopsis

PSECT name	Allocation	PSECT No.	Attributes
. ABS .	00000000 (0.)	00 (0.)	NOPIC USR CON ABS LCL NOSHR NOEXE NORD NOWRT NOVEC BYTE
ABS	00000000 (0.)	01 (1.)	NOPIC USR CON ABS LCL NOSHR EXE RD WRT NOVEC BYTE
RODATA	00000084 (132.)	02 (2.)	NOPIC USR CON REL LCL NOSHR NOEXE RD NOWRT NOVEC LONG
RWDATA	000001B0 (432.)	03 (3.)	NOPIC USR CON REL LCL NOSHR NOEXE RD WRT NOVEC LONG
SATSSS41	000003F9 (1017.)	04 (4.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC BYTE

Performance indicators

Phase	Page faults	CPU Time	Elapsed Time
Initialization	35	00:00:00.11	00:00:00.31
Command processing	135	00:00:00.71	00:00:01.40
Pass 1	269	00:00:07.47	00:00:14.13
Symbol table sort	0	00:00:00.73	00:00:00.99
Pass 2	107	00:00:01.81	00:00:02.45
Symbol table output	13	00:00:00.08	00:00:00.13
Psect synopsis output	3	00:00:00.03	00:00:00.04
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	564	00:00:10.95	00:00:19.47

The working set limit was 1500 pages.
39263 bytes (77 pages) of virtual memory were used to buffer the intermediate code.
There were 30 pages of symbol table space allocated to hold 473 non-local and 23 local symbols.
515 source lines were read in Pass 1, producing 23 object records in Pass 2.
42 pages of virtual memory were used to define 32 macros.

Macro library statistics

Macro library name	Macros defined
_255DUA28:[SHRLIB]UETP.MLB;1	8
_255DUA28:[SYS.OBJ]LIB.MLB;1	2
_255DUA28:[SYSLIB]STARLET.MLB;2	19
TOTALS (all libraries)	29

864 GETS were required to define 29 macros.

There were no errors, warnings or information messages.

```
MACRO/LIS=LIS$:SATSSS41/OBJ=OBJ$:SATSSS41 MSRC$:SATSSS41/UPDATE=(ENH$:SATSSS41)+EXECML$/LIB+SHRLIB$:UETP/LIB
```